REMARKS

Applicants respectfully request the Examiner's reconsideration of the present application as amended.

Rejected Claims under 35 U.S.C. § 103(a)

The Examiner has rejected claims 1, 3-9, 11-17, and 19-24 under 35 U.S.C. § 103(a) as being unpatentable over Andrew et al. ("Andrew"), U.S. Patent No. 5,428,403 in view of Akiwumi-Assani et al. ("Akiwumi-Assani"), U.S. Patent No. 5,532,744.

To establish *prima facie* obviousness of a claimed invention, all the claim limitations must be taught or suggested by the prior art. *In re Royka*, 490 F.2d 981, 180 USPQ 580 (CCPA 1974). (MPEP ¶ 2143.03).

Independent claim 1 of the present application includes limitations not suggested or taught by Andrew or Akiwumi-Assani. As a result, claim1 is patentable over Andrew in view of Akiwumi-Assani.

Specifically, claim 1 of the present application includes the elements of "A method for decoding compressed video comprising: reading a stream of compressed video into memory, said video having multiple pictures, each picture having one or more *independent slices;* assigning, via a first processor of a group of processors sharing said memory, *at least one independent slice per processor to be decoded by the processors in parallel.*"

However, there is no teaching or suggestion in Andrew or Akiwumi-Assani of "A method for decoding compressed video comprising: reading a stream of compressed video

into memory, said video having multiple pictures, each picture having one or more *independent slices;* assigning, via a first processor of a group of processors sharing said memory, *at least one independent slice per processor to be decoded by the processors in parallel.* "

Andrew discloses an embodiment to *encode* digitized video images for storage and reproduction (Andrew, Column 5, line 10- 12). However, applicants claim a method to *decode* compressed video as in claim 1. Also, Andrew discloses a method for estimating motion vectors for *blocks* of pixels, which is used to improve compression of motion picture data. (Andrew, Column 1, line 5-15). On the other hand, applicants claim a method in which at least one *independent slice (not motion vectors for blocks of pixels)* per processor is to be decoded (not encoded) as in claim 1.

Furthermore, Andrew does not disclose assigning via a first processor of a group of processors sharing said memory *at least one independent slice per processor to be decoded by the processors in parallel,* instead Andrew discloses partitioning the *blocks* of a frame into their horizontal rows, and *all the blocks* of a row are processed in left to right order by a *single DSP.* (Andrew, column 7, line 55-58.)

The examiner noted that Andrew "fails to disclose parallel processing the bit stream in independent units corresponding to slices" (Office Action of 4/29/02, P3)

In sum, Andrew discloses *encoding* digitized video images by which blocks are divided into their horizontal rows and all the blocks of a row are processed in left to right order by a *single* DSP. Thus, Andrew does not disclose "A method for decoding compressed video comprising: reading a stream of compressed video into memory, said video having multiple pictures, each picture having one or more *independent slices;* assigning, via a first

processor of a group of processors sharing said memory, *at least one independent slice per processor to be decoded by the processors in parallel.* "

Similarly, Akiwumi-assani does not disclose "A method for decoding compressed video comprising: assigning, via a first processor of a group of processors sharing said memory, *at least one independent slice per processor to be decoded by the processors in parallel.*" Akiwumi-assani discloses decoding digital video bit stream using a decoder block, which is made up of a plurality of decoder modules, (Akiwumi-assani, column 4, line 3-5), that utilizes the MPEG layered approach, (Akiwumi-assani, column 2, line 40-60), by which a system controller detects MPEG slice start codes that indicate the beginning of each portion of the picture area, (Akiwumi-assani, column 5, line 5-10). When the system controller detects a slice layer start code it routes it to the slice parser, which *decodes* the slice start code to determine the display position of the particular slice within a frame, (Akiwumi-assani column 5, line 35-39). That is decoding occurs initially at the slice parser using slice start code not as applicants claim "at least one independent slice per processor to be decoded *by processors (not slice parser)* in parallel."

Furthermore, Akiwumi-assani discloses that once the start code of a slice is located, the slice parser begins to count the number of bits and the number of slices associated with the first slice and subsequent slices *until either the initial number of slices per decoder module is reached or until the total number of bits is less than or equal to a set amount,* up until this point the bits are routed to a first decoder module, (Akiwumi-assani column 5, line 50-60). Meanwhile, the other decoder modules are not utilized (i.e. are not decoding) until the above detailed process is completed at the slice parser. Therefore, Akiwumi-assani does not disclose that at least one independent slice per processor is decoded by the *processors* in parallel, instead Akiwumi-assani discloses a slice parser that decodes slice start codes and then directs bits to a first decoder module until a limit is reached, which is that either the maximum number of *slices (not independent slice)* per decoder module is reached or a set

amount of bits per decoder module is reached, and only then does the slice parser directs the subsequent bits to a second module (Akiwumi-assani column 5, line 50-60).

As such, claim 1 of the present application is patentable over Andrew in view of Akiwumi-assani because it includes limitations not suggested or taught by Andrew or Akiwumi-assani.

Independent claim 9 of the present application includes limitations not suggested or taught by Andrew or Akiwumi-assani. As a result, claim 9 is patentable over Andrew in view of Akiwumi-assani.

Specifically, claim 9 of the present invention includes the limitation of "assigning, via a first processor of a group of processors sharing said memory, *at least one independent slice per processor to be decoded by the processors in parallel.*"

However, for the same reasons discussed above in so far as claim 1, there is no teaching or suggestion in Andrew or Akiwumi-assani of "assigning, via a first processor of a group of processors sharing said memory, *at least one independent slice per processor to be decoded by the processors in parallel.*"

As such, claim 9 of the present application is patentable over Andrew in view of Akiwumi-assani because it includes limitations not suggested or taught by Andrew or Akiwumi-assani.

Independent claim 17 of the present application includes limitations not suggested or taught by Andrew or Akiwumi-assani. As a result, claim 17 is patentable over Andrew in view of Akiwumi-assani.

Specifically, claim 17 of the present invention includes the limitation of "said first unit of logic further assigns, via a first processor of said group of processors sharing said memory, *at least one independent slice per processor to be decoded by the processors in parallel.*"

However, for the same reasons discussed above in so far as claim 1, there is no teaching or suggestion in Andrew or Akiwumi-assani of "said first unit of logic further assigns, via a first processor of said group of processors sharing said memory, *at least one independent slice per processor to be decoded by the processors in parallel.*"

As such, claim 17 of the present application is patentable over Andrew in view of Akiwumi-assani because it includes limitations not suggested or taught by Andrew or Akiwumi-assani.

Claims 3-8, 11-16 and 19-24 depend from the independent claims discussed above, and therefore include the limitations of the referenced independent claims. As a result, claims 3-8, 11-16 and 19-24 include the distinguished limitations, as discussed above, and are therefore patentable over Andrew in view of Akiqumi-assani.

Condition for Allowance

The Applicants submit that all rejections have been overcome and the present application is now in condition for allowance. If a telephone interview would in any way expedite the prosecution of this application, the Examiner is invited to contact John Ward at (408) 720-8300, x237.

Charge Deposit Account

If there are any additional charges, please charge them to our Deposit Account Number 02-2666.

Respectfully submitted,

BLAKELY, SOKOLOFF, TAYLOR & ZAFMAN LLP

Dated: August 20, 2002

John P. Ward
Reg. No. 40,216

12400 Wilshire Boulevard
Seventh Floor
Los Angeles, California 90025
(408) 720-8300

VERSION WITH MARKINGS TO SHOW CHANGES MADE

A marked-up version of the amended claims 1, 3, 4, 5, 9, 11, 12, 13, 17, 19 is provided below. Additions are indicated with "__" and deletions are indicated within "[ ]."

IN THE CLAIMS

1. (Once Amended) A method for decoding compressed video comprising:

reading a stream of compressed video into a memory, said video having multiple pictures, each picture having one or more independent [elements] slices;

assigning, via a first processor of a group of processors sharing said memory, at least one independent [element] slice per processor to be decoded by the processors in parallel. [and decoding the independent elements of the video in parallel.]

3. (Once Amended) The method of claim[2] 1, wherein assigning the independent [elements] slices includes assigning a varying number of slices to individual processors.

4. (Once Amended) The method of claim 3, wherein assigning the independent [elements] slices includes assigning a comparable work load to the processors.

5. (Once Amended) The method of claim 4, wherein assigning the independent [elements] slices includes placing in memory as a local variable, for each processor, the slices to be decoded by a respective processor.

6. The method of claim 5, wherein each slice includes at least one macroblock.

7. The method of claim 6, wherein said video is encoded in MPEG.

8. The method of claim 7, wherein the method of decoding is performed in real-time.

9. (Once Amended) A computer-readable medium having stored thereon a set of instructions, said set of instruction for decoding compressed video, which when executed by a processor, cause said processor to perform a method comprising;

reading a stream of compressed video into memory, said video having multiple pictures, each picture having one or more independent [elements] slices;

assigning, via a first processor of a group of processors sharing said memory, at least one independent [element] slice per processor to be decoded by the processors in parallel. [and decoding the independent elements of the video in parallel.]

11. (Once Amended) The computer-readable medium of claim [10] 9, wherein assigning the independent [elements] slices includes assigning a varying number of slices to individual processors.

12. (Once Amended) The computer-readable medium of claim 11, wherein assigning the independent [elements] slices includes assigning a comparable work load to the processors.

13. (Once Amended) The computer-readable medium of claim 12, wherein assigning the independent [elements] slices includes placing in memory as a local variable, for each processor, the slices to be decoded by a respective processor.

14. The computer-readable medium of claim 13, wherein each slice includes at least one macroblock.

15. The computer-readable medium of claim 14, wherein said video is encoded in MPEG standard.

16. The computer-readable medium of claim 15, wherein the method of decoding is performed in real-time.

17. (Once Amended) A Computer system comprising:

a plurality of processors;

a memory coupled to said plurality of processors;

a first unit of logic to read a stream of compressed video into said memory, said video having multiple pictures, with each picture having one or more independent [elements] slices; and

said first unit of logic further assigns, via a first processor of said group of processors sharing said memory, at least one independent [element] slice per processor to be decoded by the processors in parallel.

19. (Once Amended) The computer system of claim [18] 17, wherein said first unit of logic assigns a varying number of slices to individual processors.

20. The computer system of claim 19, wherein said first unit of logic assigns a comparable work load to the processors.

21. The computer system of claim 20, wherein said first unit of logic places in memory as a local variable, for each processor, the slices to be decoded by a respective processor.

22. The computer system of claim 21, wherein each slice includes at least one macroblock.

23. The computer system of claim 22, wherein said video is encoded in MPEG standard.

24. The computer system of claim 23, wherein system computer system decodes said video in real-time.